EDITED BY BILL TRAVIS & ANNE WATSON SWAGER

Bipolars provide stable current source

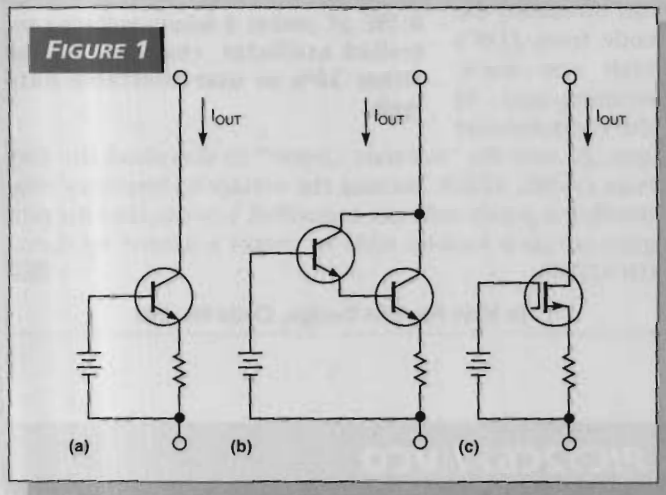
BILL MORONG, MORONG'S HARNESS, DOVER-FOXCROFT, ME

It's possible to implement a precise current source with a useful output at high frequencies, without using operational amplifiers. The circuit in **Figure 1a** suffers inaccuracies from both the V_{BE} drop and the finite base current of the transistor. The circuit in **Figure 1b** overcomes the base-current problem, but has two V_{BE} drops and does not perform well at high frequencies. The circuit in **Figure 1c** has no base-current problem and performs well at high frequencies, but is prone to inaccuracies from the V_{GS} of the FET. The circuit in **Figure 2** largely overcomes these problems.

The V_{BE} of Q_2 cancels that of Q_1. Because the base current of Q_1 diverts (via Q_3) as shown around the current-setting resistor R_1, I_{OUT} is simply two-thirds of V_{IN} divided by R_1. Because the circuit provides error cancellation, the values and voltages are not critical, provided you match the upper and lower components. In this example, Q_1 has a beta of 50; the circuit self-adjusts as beta varies. Neon-driver transistors are a good choice for Q_1 because Q_2 and Q_3 need neither good high-frequency performance nor high output impedance. High-beta (400 in **Figure 2**) transistors are appropriate, as they minimize errors. Insofar as possible, it is desirable to have similar, high betas for Q_2 and Q_3. At high frequencies, it may be beneficial to bypass the base of Q_1 to ground. (DI #2257). EDN

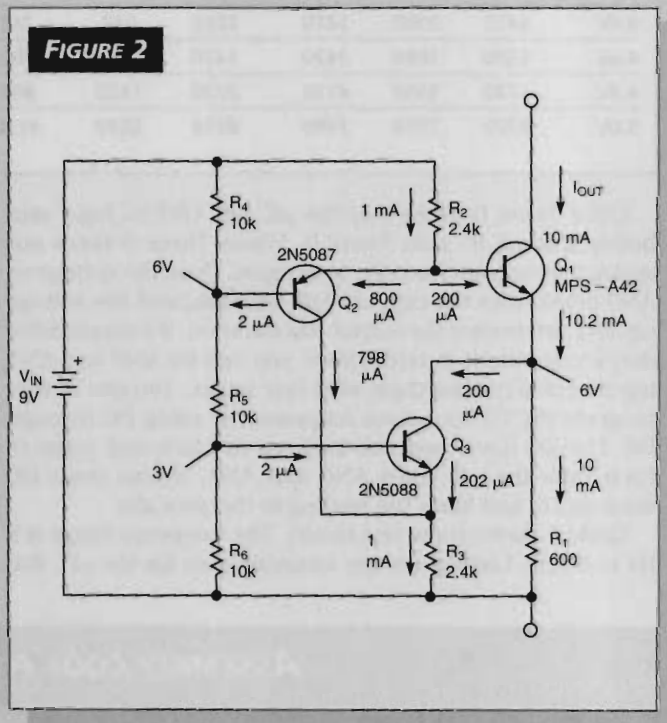
To Vote For This Design, Circle No. 386

FIGURE 1



Finite base current, temperature-dependent base-emitter drop, and gate-source-voltage variations lead to errors in single-transistor current sources.

FIGURE 2



Base-emitter-voltage and base-current cancellation are the keys to a stable, predictable current source.

8-pin µC forms one-chip programmable VCO

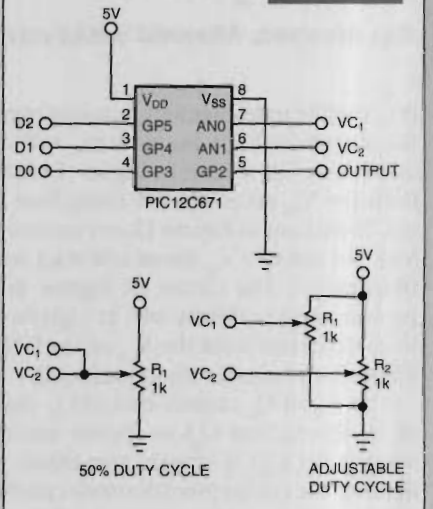
YONGPING XIA, TELDATA INC, LOS ANGELES, CA

The circuit in **Figure 1** uses a Microchip 8-pin µC (PIC12C671) as a voltage-controlled oscillator (VCO). Because the PIC12C671 has an internal 4-MHz oscillator, four-channel 8-bit A/D converters, and built-in power-reset circuitry, you need no external components to configure the VCO. The µC reads two analog inputs through AN0 and AN1. The reference voltage for the A/D conversion is the µC's power supply V_{DD}. The converted 8-bit data determines the duration of output high and output low. Assume, for example, the digitized outputs from AN0 and AN1 are 43 and 87, respectively. Timer 0 loads the 43 after the µC sets output GP2 to logic one. Timer 0 receives its timing from the internal clock.

TABLE 1—TEST RESULTS FOR PIC-BASED VCO

VC1/VC2	D2/D1/D0							
	000	001	010	011	100	101	110	111
0V	899	474	243	123	62.3	31.2	15.6	7.9
0.2V	928	490	252	127	64.4	32.3	16.2	8.1
0.4V	963	509	263	133	67.2	33.8	16.9	8.4
0.6V	1000	531	274	140	70.1	35.2	17.6	8.8
:	:	:	:	:	:	:	:	:
1.0V	1090	579	300	153	77.3	38.7	19.5	9.7
:	:	:	:	:	:	:	:	:
2.4V	1560	858	451	232	118	59.5	29.7	14.9
:	:	:	:	:	:	:	:	:
4.4V	4410	2980	1810	1010	537	268	141	71.1
4.6V	5320	3880	2420	1410	768	402	206	104
4.8V	6720	5550	4130	2730	1470	807	474	243
5.0V	8220	7970	7490	6710	5550	4130	2720	1620

FIGURE 1



A PIC μC makes a handy voltage-controlled oscillator, configurable for either 50% or user-selectable duty cycle.

Once Timer 0 times out, the μC sets GP2 to logic zero before loading 87 into Timer 0. When Timer 0 times out again, the program starts the loop again. Thus, the voltage on AN0 determines the output-high duration, and the voltage on AN1 determines the output-low duration. If a simple 50%-duty-cycle output is satisfactory, you can tie AN0 and AN1 together and control them with one source. You can further program the VCO's output frequency by using D0 through D2. The μC has a prescaler between its clock and Timer 0. Each time the μC reads AN0 and AN1, it also reads D0 through D2 and loads the reading to the prescaler.

Table 1 shows some test results. The frequency range is 8 Hz to 8 kHz. **Listing 1** is the assembly code for the μC. You can download the code from *EDN*'s Web site www.ednmag.com. At the registered-user area, go into the "Software Center" to download the files from DI-SIG, #2259. Because the voltage-to-frequency relationship is purely software-controlled, you can alter the program or use a look-up table to obtain a desired v-f curve. (DI #2259).

To Vote For This Design, Circle No. 387

ASSEMBLY CODE FOR PIC12C671 VCO

```
TMR0        equ     0x01
STATUS      equ     0x03
GPIO        equ     0x05
INTCON      equ     0x0b
PIR1        equ     0x0c
ADRES       equ     0x1e
ADCON0      equ     0x1f
ADCON1      equ     0x9f
OPT         equ     0x81
port_dir    equ     0x85
high_data   equ     0x21
low_data    equ     0x22
out_status  equ     0x23
temp        equ     0x24

            org     0x0             ;
            goto    main            ;
                                    ;
            org     0x4             ;
            movf    GPIO, 0         ;
            movwf   temp            ;
            rrf     temp, 1         ;
            rrf     temp, 1         ;
            rrf     temp, 1         ;
            movf    temp, 0         ;
            andlw   0x07            ;
            addlw   0x80            ;
            option                  ;
            btfss   out_status, 0   ;
            goto    set_high        ;
            goto    set_low         ;
set_high                            ;
            bsf     out_status, 0   ;
            bsf     GPIO, 2         ;
            movlw   0x41            ; A/D input from AN0
            movwf   ADCON0          ;
            call    delay           ; data acquisition
            bsf     ADCON0, 2       ; start A/D conversion
loop_2      btfsc   ADCON0, 2       ;
            goto    loop_2          ;
            movf    ADRES, 0        ;
            movwf   TMR0            ;
            bcf     INTCON, 2       ; clear TMR0 interrupt flag
            retfie                  ;
                                    ;
delay                               ;
            movlw   0x03            ;
            movwf   temp            ;
dly_lp      decfsz  temp, 1         ;
            goto    dly_lp          ;
            return                  ;
                                    ;
main                                ;
            clrf    GPIO            ;
            bsf     STATUS, 5       ; bank 1
            movlw   0x85            ;
            movwf   OPT             ;
            movlw   0x0b            ;
            movwf   port_dir        ;
            movlw   0x04            ;
            movwf   ADCON1          ;
            bcf     STATUS, 5       ; bank 0
            movlw   0xa0            ;
            movwf   INTCON          ;
loop                                ; endless loop
            goto    loop            ;
                                    ;
            end
```

Cable tester is fast and cheap

ABEL RAYNUS, ARMATRON INTERNATIONAL, MELROSE, MA

The cable tester in **Figure 1** uses a low-end 8-bit µC. The specific µC to use depends on the number of conductors in the cable you want to test. For the current application, two types of cables were under test, one with three conductors and another with seven. So, the Motorola 68HC705P9 µC was suitable. The program first determines which type of cable is under test by checking the cable-switch position (**Listing 1**). Then, the program checks each conductor line from A_0/C_0 to A_7/C_7 by putting a high-level voltage from output port A on one end of the wire and measuring the response on the other end, which is connected to the input port C. If all of the checks show conductivity, the green "pass" LED turns on. In the opposite case, the red "fail" LED turns on. The test checks not only for conductivity but also for the presence of a cross connection.

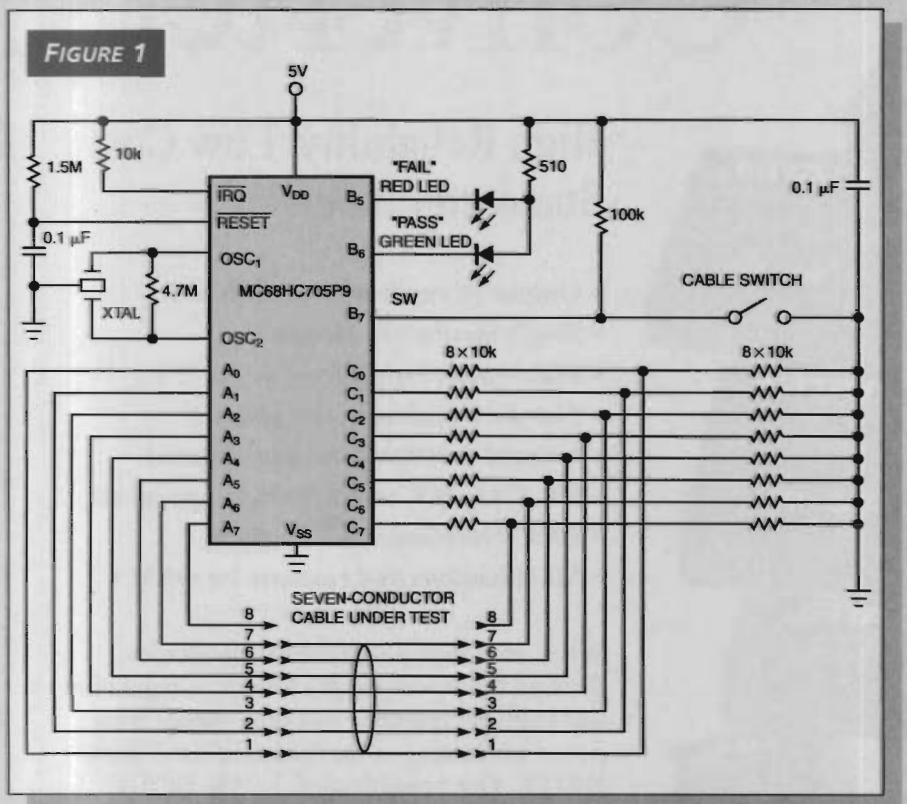
If you want to test a variety of cables, you can use more switches. If a cable has more than eight conductors, you can use a different type of µC or multiplex the inputs.

Listing 1 and an assembly-language program are available for downloading from *EDN*'s Web site, www.ednmag.com. At the registered-user area, go into the Software Center to download the file from DI-SIG, #2225.(DI #2225)

EDN

To Vote For This Design, Circle No. 388

FIGURE 1



This simple cable tester verifies that the signal from port A appears at port C of the µC.

LISTING 1—SPICE MODEL FOR DEAD-TIME GENERATOR

```
.nolist
$include "STD_P9.asm"
.list
* I/O PORTS
SW   equ  7 ;prtB
GRN  equ  6 ;prtB
RED  equ  5 ;prtB
* CONSTANTS
W1   equ  7 ;7 conductors cable
W2   equ  3 ;3 conductors cable
* VARIABLES
     org  RAM
W    rmb  1 ;wire register
N    rmb  1 ;wire counter
* INITIALIZATION
     org  MOR
     fcb  $00 ;no watchdog
     org  ROM
init lda  #%11111111
     sta  ddrA
     lda  #%01111111
     sta  ddrB
     clr  prtC
main brclr SW,prtB,m1 ;which cable is under test?
     lda  #W1
     sta  W
m2   clr  N
     ldx  #1
start inc  N
     stx  prtA
     txa
     eor  prtC
     bne  FAIL
     lda  N
     cmp  W
     beq  PASS
     aslx
     bra  start
m1   lda  #W2
     sta  W
     bra  m2
FAIL bset GRN,prtB ;green LED off
     bclr RED,prtB ;red LED on
     bra  main
PASS bclr GRN,prtB ;green LED on
     bset RED,prtB ;red LED off
     bra  main
     org  $1ffe
     fdb  init     ;restart address
.end
.nolist
```

Capacitive sensor "likes" parasitics

BORIS KHAYKIN, CANDID LOGIC INC, MADISON HEIGHTS, MI

Stray capacitance is a common problem with capacitive sensors. The capacitance changes within the measurement range are normally much smaller than the strays; the result is a loss of sensitivity. Various methods are available to increase the relative sensitivity ($\Delta f/f_0$): frequency subtraction, the use of bridges, and the use of a negatron to subtract the strays, for example. The idea here is not to do battle with the stray, but rather use it and turn its drawbacks to your advantage. This method uses frequency-dependent hysteresis in a classic op-amp multivibrator. **Figure 1** shows a simple, flexible design for a capacitive sensor.

Without capacitor C_2, the design is a classic multivibrator based on comparator IC_1 with output buffer IC_2. If $R_1=R_2$, the frequency is

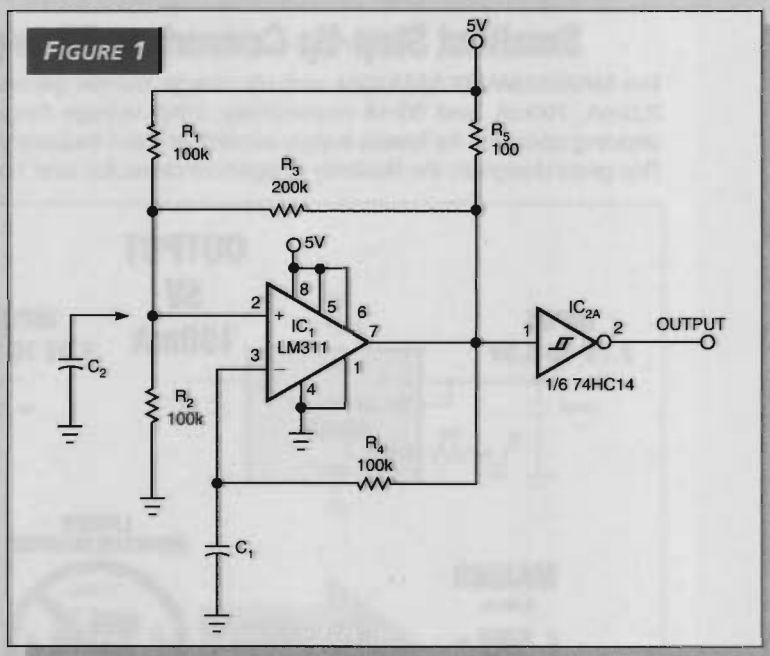
$$f=\frac{1}{2R_4C_1\ln\left(1+\frac{R_2}{R_3}\right)}.$$

R_1, R_2, and R_3 define the hysteresis, 900 mV with the values shown. Frequency (f) is a function of capacitor C_1, as **Figure 2** shows. Without C_2 and with C_1=60 pF and ΔC=20 pF, f_0=159 kHz and the relative sensitivity $\Delta f/f_0$ is –18%. With C_2 connected in parallel with R_2, the hysteresis becomes frequency-dependent. The capacitive reactance ($X_C=1/2\pi fC$) in parallel with R_2 reduces the hysteresis in an inverse proportion to the frequency. As a result, the frequency increases. This increase reduces X_C, further reduces the hysteresis, and leads to a further increase in frequency. Thus the relative sensitivity $\Delta f/f_0$ increases significantly (see **Figure 2** with C_2=40 pF).

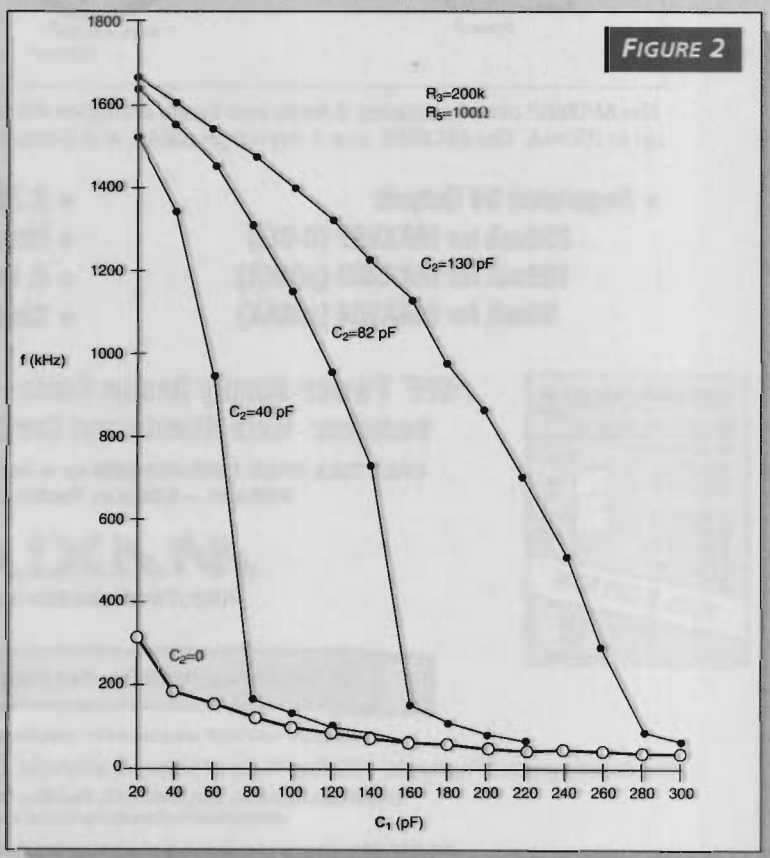
With C_2=40 pF, C_1=60 pF, and ΔC=20 pF, f_0=945.5 kHz and the relative sensitivity ($\Delta f/f_0$) is –82%. The sensitivity ($\Delta f/f_0$) (38.6) in this case is 26 times as high as the case without C_2 ($\Delta f/f_0$=1.45). You can obtain even more interesting results by replacing C_2 with a sensing capacitor. If C_1=200 pF, changing the value of C_2 from 0 to 200 pF changes the hysteresis from 900 to 28 mV, and changes the frequency from 30 to 1300 kHz. **Figure 3** shows output frequency (f) as a function of capacitance C_2. With C_2=100 pF and ΔC=20 pF, f_0=145.2 kHz and the relative sensitivity ($\Delta f/f_0$) is +393%. Thus, the frequency is directly proportional to the capacitance.

As **Figure 3** demonstrates, you can adjust the desired initial frequency with R_3, and the sensitivity with R_5. Note that the higher sensitivity in this

The addition of "stray" capacitance to Figure 1's circuit significantly increases the sensor's sensitivity.



If you can't beat 'em, join 'em. This circuit exploits stray capacitance to increase its own sensitivity in making capacitance measurements.

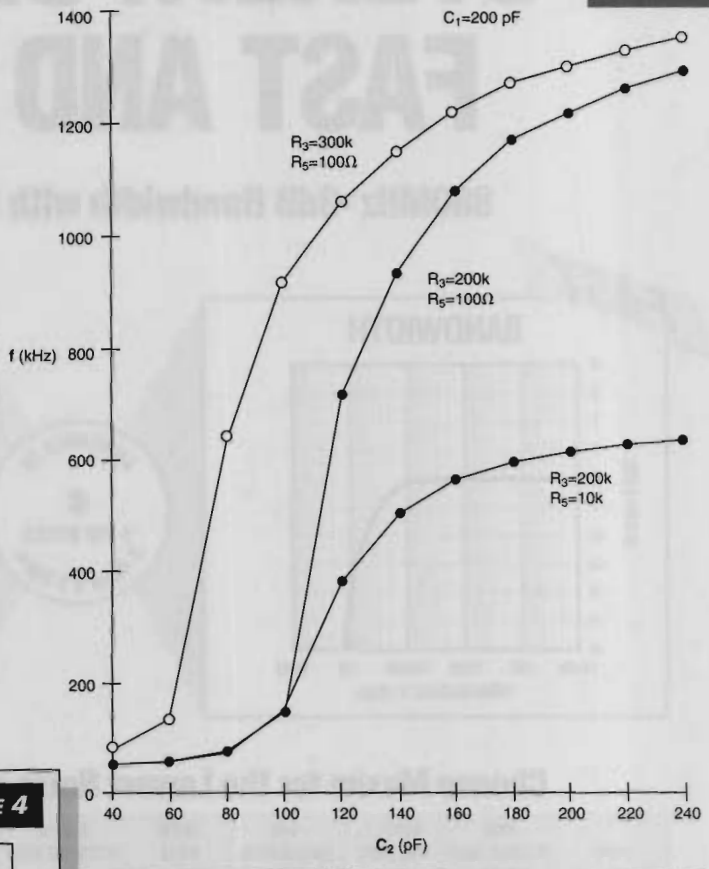


example occurs with a significant stray capacitance (100 pF, for example). If the real sensor has lower initial capacitance (50 pF, for example) the simple addition of a 50-pF capacitor in parallel with the sensor increases the sensitivity. The sensor "likes" the stray capacitance as it produces frequency-dependent hysteresis that, in turn, provides higher sensitivity. You could also use the added capacitor for temperature compensation.

If you use an extremely fast op amp or comparator in this design, there is a certain value of C_2 for which the output frequency jumps up a few kHz with a hysteresis of 5 to 7 pF (**Figure 4**). This quirk is particularly useful in the design of supersensitive capacitive switches. You can adjust the switching point with R_3 and/or a capacitor in parallel with C_2. You can adjust the hysteresis by using a small resistance connected in series with C_2. On the other hand, the use of a slower comparator linearizes the frequency-versus-capacitance characteristic. For example, test results show that with an LM319 comparator, R_3=200 kΩ, R_5=200Ω, and C_1=200 pF, the output frequency follows the empirical equation f=140+3.327(C_2–100) kHz with 3% nonlinearity within the range C_2=100 to 400 pF. (DI #2258). EDN

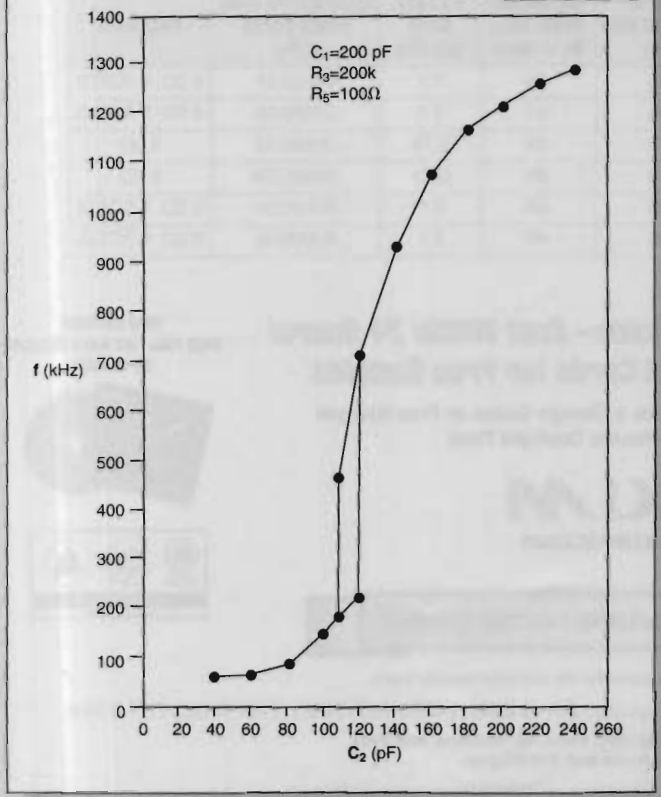
To Vote For This Design, Circle No. 389

FIGURE 3



This response is the result of using the "stray" capacitance as the sensing element in the Figure 1's circuit.

FIGURE 4



An idiosyncrasy inherent in fast op amps or comparators produces an abrupt jump in frequency for a small change in capacitance.